Runtime Reconfigurable Beamforming Architecture for Real-Time Sound-Source Localization

Bruno da Silva, Laurent Segers, An Braeken and Abdellah Touhafi
Dept. of Industrial Sciences (INDI), Vrije Universiteit Brussel (VUB), Brussels, Belgium

***Abstract*—Sound-source localization is used in many different real-time acoustic applications. Microphone arrays have the potential capability to recognize, profile and locate sound-sources in noisy environments. The quality response of such sensor arrays, however, is determined by the quantity of microphones. A higher number of microphones increases the computational demand, making real-time response challenging. In this paper, we present a scalable and runtime reconfigurable architecture to provide accurate sound-source localization in real-time. On one hand, the reconfigurable architecture is designed to be scalable in order to support a variable number of microphones. On the other hand, we use runtime reconfigurable look-up tables (CFGLUTs) to provide a dynamic response in real-time. Experiments demonstrate how an accurate sound-source localization is obtained in less than a few hundred milliseconds. As far as we are aware, it is the first time that runtime reconfiguration is applied to a reconfigurable architecture consisting of a sensor array.**

I. Introduction

Microphone arrays are becoming popular to many different applications. The localization of sound-sources is applied in the environmental, industrial and military domains. For instance, military applications usually range from localizing sniper fire to identify noisy engine parts. The time response becomes crucial for those vital situations.

Due to the recent miniaturization and price drop of good quality microphones in the form of Microelectromechanical systems (MEMS), microphone arrays are rapidly adopting such type of microphones. MEMS microphones are much cheaper than traditional microphones and have a relatively good signal-to-noise ratio (SNR) and frequency response. Their small size allows miniature arrays that are only several centimeters in diameter with a high level of integration.

Most of the signal processing demanded by such arrays are traditionally computed in general-purpose processors. The computational demand, however, is directly related to the number of microphones of the array, which is intensely increasing thanks to the low-cost MEMS technology. Thanks to the high-computational power and low latency response that FPGAs offer nowadays, we believe that FPGAs are not only able to manage relatively large microphone arrays but they also offer a faster response.

With the additional number of microphones and real-time applications in mind, we propose a flexible, scalable and runtime reconfigurable architecture able to satisfy the most time demanding sound-source localization applications. Our main targets are the scalability of the system and the capability to dynamically reconfigure the microphone array in order to detect sound-sources in real-time ($< 200ms$).

The main contributions of this work can be summarized as follows:

- A complete FPGA implementation for a real-time sound-source detection and location.
- A scalable and modular architecture which supports variable number of microphones.
- A runtime reconfiguration of the architecture to dynamically respond to the acoustic context and to accelerate the sound-source location.

This paper is organized as follows. Section II presents related work. The description of a modular MEMS microphone array is done in Section III. Our scalable and reconfigurable architecture is presented in Section IV. In Section V the proposed architecture is evaluated. Finally, the conclusions are drawn in Section VI.

II. Related Work

The use of microphone arrays for sound-source localization is a well-researched problem. Many microphone arrays are designed for a specific type of sound-sources, which characteristics are known in advance. An example of a military application is a counter-sniper system [1], which based on the sound-source profile and triangulation, is able to locate the sniper's position.

In many situations the sound-source characteristics are unknown and need to be blindly detected and located. In [2] the authors propose a beamforming-based acoustic system composed of up to 33 MEMS microphones for localization of the dominant noise source. The authors in [3] describe the design on an FPGA of an eight-element digital MEMS microphone array for distant speech recognition.

Our proposed architecture is a revised version of the SoundCompass presented in [4]. The new SoundCompass uses a particular runtime reconfigurable 5-input LUT (CFGLUT5) [5] to dynamically adapt its behavior based on the sound-source detection. This component has been available for several years but has not widely been adopted.

III. Microphone Array Description

The SoundCompass is designed for far-field and non-diffuse sound fields. The array is composed by 52 ADMP521 MEMS microphones in a 20-cm circular printed board (PCB). The planar microphone array geometry is composed of four concentric sub-arrays of 4, 8, 16 and 24 MEMS microphones. Each sub-array is differently positioned in order to facilitate the capture of the spatial acoustic information. This allows to not only

perform spatial sampling of the surrounding sound field using beamforming techniques, but also to dynamically modify the sensor array response by individually activate or deactivate sub-arrays. On one hand, using beamforming the array is focused in one specific direction or orientation, by amplifying all sound coming from that direction and by suppressing sound coming from other directions. A continuous steering of the focus direction in a 360°sweep allows to measure the variations of the surrounding sound field in all directions. On the other hand, the distributed geometry of the MEMS microphones allows to adapt the sensor to different sound-source profiles. Therefore, only a few number of sub-arrays may be active, decreasing the computational requirements and becoming more power efficient.

The ADMP521 MEMS microphones are selected due to their omnidirectional polar response and a wide-band frequency response ranging from 100 Hz up to 16 kHz [6]. These digital MEMS microphones generate multiplexed pulse density modulation (PDM) as output using a digital to analogue converter based on a sigma-delta converter. Most microphones typically use a fourth order sigma-delta converter, which utilizes an embedded integrator-comparator circuit to sample analog signals and outputs a 1-bit signal. As side effect, sigma-delta converters reduce the added noise in the audio frequency spectrum by shifting it to higher frequency ranges. Furthermore, in order to have sufficient audio quality, the DAC typically oversamples the audio signal with a 64-factor [6]. The ADMP521 MEMS microphones need a clock in a 1 to 3 MHz range to oversample the audio signal and to generate the PDM output signal. Consequently, the PDM signal needs not only to be filtered but also to be downsampled to retrieve the audio signal in a Pulse-Code Modulation (PCM) format.

The beamforming technique applied in the SoundCompass is a widely adopted Delay-and-Sum beamforming [7]. Although Delay-and-Sum beamforming assumes a fixed number of microphones and a fixed geometry, our scalable solution satisfies those restrictions while offering a flexible geometry. This beamforming is applied to a discrete number of orientations or angles, which determines the angular resolution of the microphone array. The processed measurements are represented in a polar power plot or polar map, showing the output power (P) when pointing all directions. The characteristics of the main lobe when considering a single sound-source scenario determines the directivity (D_p) of the microphone array, which can be used to metric the quality of the array [8].

IV. Architecture Description

Figure 1 depicts the main components of the proposed architecture. The architecture is designed to operate in streaming fashion in order to achieve the fastest possible response. The initial latency is partially determined by the maximum order of the filters, since they need to be reset for every orientation. The first filtered values are not generated before additional clock cycles due to the decimation factors of the filters, which enlarges the initial latency and provides extra clock cycles for a fast fine-grain reconfiguration.

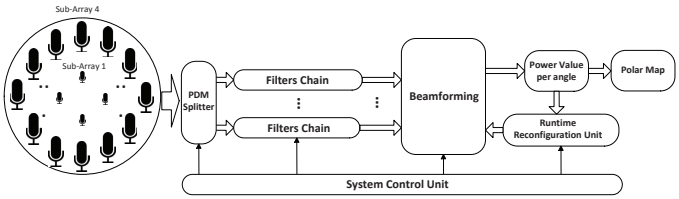

Fig. 1. *Main blocks of the scalable and reconfigurable architecture.*

Our architecture uses a 2MHz clock for the ADMP521 microphones and for the functional frequency of the design. The architecture is forced to operate at this rate since the inputs can not be obtained faster.

Every pair of microphones have their PDM output signal multiplexed in time. A PDM splitter block demultiplexes the PDM input signal to obtain the individual sampled data from each microphone.

The next stage consists of a cascade of filters to downsample and filter each microphone signal. Unfortunately, the filters cannot be time-multiplexed while processing each microphone. Firstly, a Cascaded Integrated-Comb (CIC) filter is implemented to reduce the signal bandwidth and to remove the higher frequency noise. This type of low pass filter only involves additions and subtractions [9]. The CIC filter is followed by a 16th order low-pass compensation FIR filter designed in a serial fashion in order to reduce the resource consumption. The CIC filter generates an output each 16 clock cycles due to its decimation factor. Therefore, the FIR filter has 16 clock cycles to compute each input value, which determines its maximum order. The filtered signal is then further decimated by a factor 4 to obtain a 32 khz audio signal, which satisfies the Nyquist theorem since we target frequencies below 12kHz. The last component of the filter chain block is a running average, which is used to cancel out the effects caused by the microphones DC offset output which would lead to a constant offset in the beamformed values.

The next stage is the Delay-and-Sum beamforming block. It receives the filtered PDM input signal for the target frequency range, without the DC noise and downsampled by a factor of 64. The search for the sound sources is possible by continuously steering loop in 360°. In our case, the number of supported orientations is limited to 64, which represents an angular resolution of 5.625 degrees. A higher number of orientations would increase the angular resolution, but would not only demand a larger execution time per steering loop but also more FPGA's memory resources to store the pre-computed delays.

Once the filtered data has been properly delayed and added for a particular orientation θ, $P(\theta)$ is computed at the time-domain to conform the polar map.

The main target of this proposed architecture is its adaptability to the dynamic behavior of acoustic environments. The architecture needs to be scalable, in order to support a variable number of MEMS microphones while offering a fast and accurate response to trigger spontaneous events.

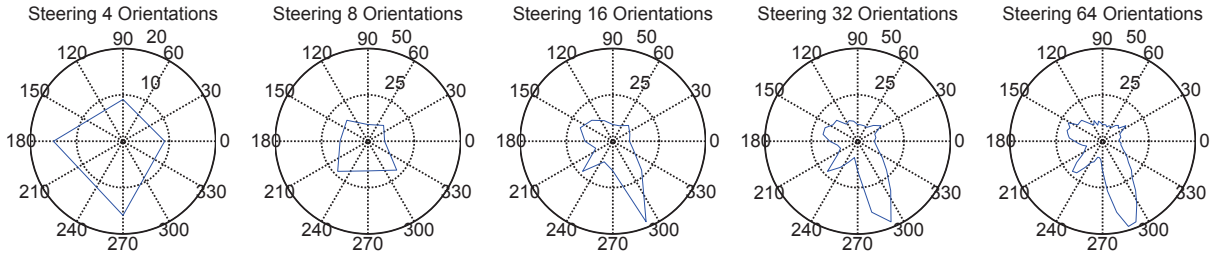

Fig. 2. Polar maps with different angular resolution locating a sound-source of 8 kHz. A low number of orientations leads to wrong sound-source location.

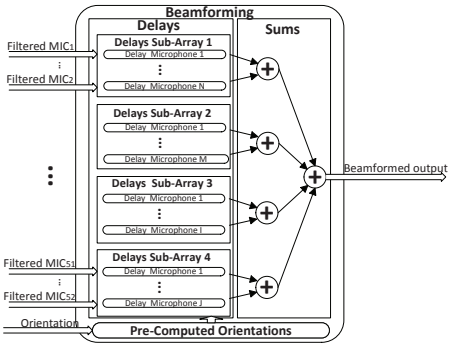

Fig. 3. *Details of the internal structure of the proposed modular Delay-and-Sum beamforming.*

A. A Scalable Architecture

One of the features of the ADMP521 microphones is their low-power sleep mode when no clock signal is provided. Our architecture allows to use the clock signal to activate or deactivate sub-arrays, decreasing the overall computational demand and power consumption. Therefore, despite the number of filter chain blocks must be the same as the number of microphones, the unnecessary filter chain blocks can be just disabled.

The beamforming stage (Figure 3) depends on the number of microphones and sub-arrays. All possible delay values are pre-computed, grouped based on which orientation correspond and stored in BRAMs during compilation time. In order to support a variable number of microphones, the filtered microphone signals are grouped following their sub-array structure. Thus, instead of implementing one simple Delay-and-Sum of 52 microphones, there are four Delay-and-Sum operations in parallel for the 4, 8, 16 and 24 microphones. Consequently, only the Delay-and-Sum beamforming block linked to an active sub-array is enabled. The disabled beamformers are set to zero in order to avoid any negative impact on the beamforming operation.

B. Runtime Reconfiguration

Figure 2 depicts how an incremental number of orientations leads to an accurate sound-source location. This resolution, however, has a significantly high timing cost. A couple of strategies are proposed to accelerate the power peak detection, and therefore, the sound-source. The first strategy consists to reduce the beamforming exploration to only 8 orientations with an angular separation of 45 degrees. Once a steering loop

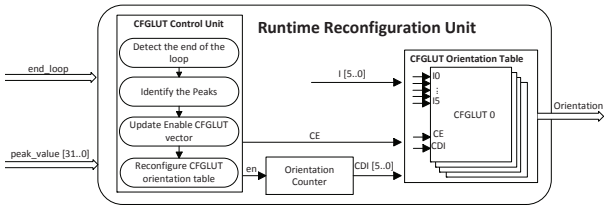

Fig. 4. *Runtime Reconfigurable Unit which adapts the orientations for the beamforming based on the previous sound-source detections.*

ends, the orientations are rotated one position, that represents a shift operation in the pre-computed orientation table. In the worst case scenario, all the supported 64 orientations are monitored after 8 steering loops.

The rotated or shifted strategy can be combined with a runtime reconfiguration. The orientation of the peak detected is used to double the angular resolution in that particular direction in order to monitor the neighbouring orientations. If the detected peak remains at the same orientation, the resolution for that direction is continuously doubled. Otherwise, the resolution increases for the orientation of the new peak while decreasing for the previous one. Due to the nature of the main lobe the reconfigurable shift strategy converge to the maximum peak. We propose the use of CFGLUTs to implement this dynamic behavior of the sound-source localization. The main advantage of the CFGLUTs is their fast reconfigurability since it only demands few clock cycles. This is a critical factor in our proposed architecture, since the fast response of the SoundCompass is one of the main quality factors.

Figure 4 depicts the main components of the Reconfigurable Unit block. The CFGLUT Control Unit monitors the steering search and detects when a whole loop is completed. In the meantime, the polar map is generated and the peak of the output power is detected in a particular orientation. This orientation is used to update the reconfigurable vector, which is used to fetch the CE signal of the CFGLUTs. This reconfigurable vector determines what orientations are going to be active in the new steering loop. Once a loop is completed, the enable vector is calculated and used as mask to reconfigure the CFGLUT Orientation table. Therefore, only the previous peak, the orientations closed to the new peak and the 8 shifted initial orientations are considered in the new steering loop. Finally, an index pointing to the initial value of the CFGLUT Orientation table is used by the System Control Unit to initialize the new

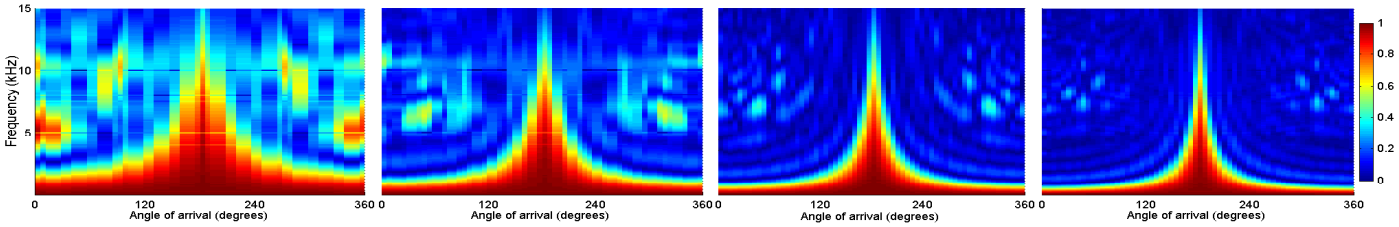

Fig. 5. *The waterfall diagrams show the normalized power output of the combined sub-arrays in all directions for all frequencies. The left two figures depicts the frequency response of the inner sub-array and by combining the two most inner sub-arrays. The middle figure shows the response by combining with sub-array 3, which is composed by 16 microphones. The right figure shows the frequency response when combining all sub-arrays. The diagrams provide a clear indication of how the array becomes more directive as the frequency increases.*

Array Configurations	One Orientation	64 Orientations	Average Shift Strategy	Average Reconfigurable Shift Strategy
Sub-array 1	6.165 ms	394.560 ms	104.805 ms	73.980 ms
Sub-arrays 1 & 2	6.294 ms	402.753 ms	106.998 ms	75.528 ms
Sub-arrays 1 & 2 & 3	6.421 ms	410.945 ms	109.157 ms	77.052 ms
All Sub-arrays	6.550 ms	419.144 ms	111.350 ms	78.600 ms

TABLE I
The execution time of the proposed peak detection strategies are compared when combining sub-array.

search. The number of clock cycles required depends of the filter order, but part of this resetting time can be also dedicated to reconfigure the CFGLUT Orientation table. Therefore, the reconfiguration cost is reduced to few clock cycles.

V. Results

The experimental platform is a Digilent Zedboard and an FMC-XM105 mezzanine card, which is interconnected to the microphone array. The output polar map is transmitted to a host through UART. A Matlab simulator, which mimics the RTL implementation, has been developed in parallel in order to validate the experimental results.

Figure 5 depicts the influence of the different microphone sub-array geometry in the frequency response of the array. This frequency response has a strong variation at the main lobe, and therefore, in D_P. We consider a threshold of 8 for D_P, which corresponds to $\frac{1}{8}$ of the unit circle. In case sub-array 1 is used separately, prominent peaks at different angles at different frequencies diminish the ability of the method to clearly locate the main lobe. Only decreasing the threshold to 4 it is possible to clearly start detecting sound-sourced at a frequency of 3kHz. Considering $D_P = 8$, the minimum detectable sound-source frequency starts at 2.4kHz while combining sub-arrays 1, 2 and 3 and all sub-arrays reaches 2.2 kHz and 1.8kHz respectively. The study of the frequency response of the microphone array confirms that, to achieve the best frequency response, the combination of sub-arrays to use as many microphones as possible is preferable in most of cases.

The execution time for each configuration and for the different peak detection strategies are summarized in Table I. The combination of multiple sub-arrays sligthly increases the execution time per orientation due to the decomposed Delay-and-Sum beamforming. The differences between the microphones of two sub-arrays introduce additional delays in the beamforming in order to properly delay the filtered signals. The exploration of the 64 available orientations demands almost half of a second when all the sub-arrays are considered. Nonetheless, either proposed strategy offer a significant acceleration while locating the sound-source. The average number of orientations are obtained from experimental measurements and they are 16.225 and 11.125 for the shift strategy and the reconfigurable shift strategy respectively. The results demonstrates that the sound-source can be located in real-time.

VI. Conclusion

In this paper we propose a flexible, scalable and runtime reconfigurable architecture for sound-source localization. On one hand, the scalable approach in our design allows to vary the microphone array response based on performance or power efficiency. On the other hand, the supported runtime reconfiguration using CFGLUTs offers an unique chance to dynamically adapt the array response to the variable acoustic environment in only few clock cycles.

VII. Acknowledgements

This work is a result of the CORNET project "DynamIA: Dynamic Hardware Reconfiguration in Industrial Applications". It is funded by IWT Flanders with reference number 140389.

References

[1] Sallai, J., et al. *Weapon classification and shooter localization using distributed multichannel acoustic sensors.* Journal of Systems Architecture 57.10 : 869-885. 2011

[2] Salom, I., et al. *An Implementation of Beamforming Algorithm on FPGA Platform with Digital Microphone Array.* In Audio Engineering Society Convention 138. Audio Engineering Society. 2015.

[3] Zwyssig, E., et al. "A digital microphone array for distant speech recognition." Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on. IEEE, 2010.

[4] Tiete, J., et al. *SoundCompass: a distributed MEMS microphone array-based sensor for sound source localization.* Sensors, 14(2), 1918-1949. 2014.

[5] Xilinx, Inc., *Xilinx Virtex-5 Libraries Guide for HDL Designs*, Aug. 2009.

[6] AnalogDevices. *ADMP521 datasheetUltralow Noise Microphone with Bottom Port and PDM Digital Output*, Technical Report, Analog Devices: Norwood, MA, USA, 2012.

[7] Johnson, Don H., and Dan E. Dudgeon. *Array signal processing: concepts and techniques*. Simon & Schuster, 1992.

[8] Taghizadeh, M., et al. *Microphone array beampattern characterization for hands-free speech applications*, Sensor Array and Multichannel Signal Processing Workshop (SAM), 2012 IEEE 7th. IEEE, 2012.

[9] Hogenauer, E. *An economical class of digital filters for decimation and interpolation*. Acoustics, Speech and Signal Processing, IEEE Transactions on 29.2 : 155-162. 1981.